REMARKS/ARGUMENTS

Claims 1 through 18 are pending in the present application. Claims 1, 4, 9 and 18 have been amended. No new matter has been added.

Interview Summary

Applicants' undersigned representative conducted a telephonic interview of this application with Examiner Roche on November 9, 2004. During that interview, Applicant's undersigned representative discussed the present application and pending rejections with Examiner Roche. Per that discussion, Applicant's undersigned representative has amended the independent claims.

Applicants' undersigned representative wishes to thank Examiner Roche for conducting the telephonic interview and for his continued consideration of this application.

Prior Art Rejections

The Disclosed Systems and Methods

Applicants have noted that existing systems for using remotely stored file data by locally executed applications have a limitation that requires the remotely stored files to be manually downloaded before using. As described in the patent specification:

> Unfortunately, while enormously helpful, the use of data files stored on the Internet as parameters in locally executed applications (e.g., the compiler application noted above) is associated with certain drawbacks. **In particular, in order to use stored Internet files, the files must generally first be downloaded to a local storage medium. Such downloading generally requires a separate step manually initiated by the developer or other software user.**
> For example, returning to the example described above, since compilers typically do not accept URL specified files as parameters, in order to compile the source and header files atom.c and atom.h, the developer must first copy the files to a local directory (e.g., via an FTP protocol utility). Only when the source and header files have been downloaded can the developer compile the shared source code (e.g., by causing execution of a compile command such as "cl /FeMyApp.exe main.c atom.c"). Furthermore, to ensure the latest version of the CII Atom interface is used, the developer must complete

such a download each time the interface is needed for a given
compilation, resulting in significant inconvenience.

(Specification, p. 2. ln. 3 – p. 3, ln. 1) (emphasis added).

Accordingly, Applicants have disclosed systems and methods wherein the conversion

of the source code file into a file executable on the client computer begins before the source

code file has been completely downloaded to the client computer.

Claim 1 is directed to a method for compiling a source code file on a client computer,

the source code file being stored on a remote server computer and being assessable via web

protocols, the method comprising:

> (a)   accepting a manually specified compile command, the
> compile command including a set of parameters, the set of parameters
> including an identifier corresponding to the source code file; and
> (b)   executing a compile procedure corresponding to the
> compile command, the compile procedure effecting conversion of the
> source code file into a file executable on the client computer,
> wherein step (b) includes **downloading the source code file from
> the remote server computer to the client computer using web
> protocols without executing a manually specified download command,
> and <u>the conversion of the source code file into a file executable on the
> client computer begins before the source code file has been completely
> downloaded to the client computer,</u> and**
> further wherein the identifier corresponding to the source code
> comprises an identifier of executable code, and downloading the source
> code file comprises transmitting to the remote server the identifier
> corresponding to executable code and at least one parameter used by the
> executable code to identify the source code.

Similarly, claim 4 is directed to a computer-based method for executing an

application on a client computer, the application functioning to process file data stored on a

remote server computer, the file data stored on the remote server computer being accessible

via web protocols, the method comprising:

> (a)   accepting a manually specified execute command, the
> execute command including a set of parameters, the set of parameters
> including an identifier corresponding to the file data; and
> (b)   executing a procedure corresponding to the execute
> command, the procedure manipulating the file data on the client computer,
> wherein step (b) includes **downloading the file data from the
> remote server computer to the client computer using the web
> protocols without executing a manually specified download command,**

DOCKET NO.: MSFT-0234/155631.1
Application No.: 09/767,768
Advisory Action Dated: September 22, 2004

PATENT
REPLY FILED UNDER EXPEDITED
PROCEDURE PURSUANT TO
37 CFR § 1.116

**and the manipulation of the file data on the client computer begins
before the file data has been completely downloaded to the client
computer,** and
further wherein downloading the file data comprises transmitting
to the remote server computer an identifier of executable code and at least
one parameter used by the executable code to derive the file data.

Claim 9 is directed to a computer system including a processor, memory associated

with the processor, and a storage medium capable of storing a data file, the data file having a

corresponding file identifier, the system comprising:

(a)   an application software component comprised of
instructions in the memory and executable by the processor, the
application software component functioning to process the data file; and
(b)   an I/O software component comprised of instructions in the
memory and executable by the processor, the I/O software component
functioning to accept the file identifier, to **determine whether the file
identifier is a URL and, if so, to retrieve the data file from a remote
server using the file identifier** and, if not, to retrieve the data file from
the storage medium using the file identifier,
**wherein the application software component processing the
data file begins before the data file has been completely retrieved
from the remote server,**
wherein said file identifier identifies executable code, and
wherein said I/O software component functioning to retrieve the
data file from a remote server using the file identifier operates by
transmitting to the remote server said file identifier with at least one
parameter, said at least one parameter being executable by the executable
code identified by said file identifier.

Claim 18 is directed to a computer-readable storage medium used in a computer

system having a processor, memory associated with the processor and a storage device

having a data storage medium, the computer-readable storage medium having instructions

capable of being executed by the processor for performing the following:

(a)   accepting a file identifier corresponding to a data file;
(b)   **determining whether the file identifier is a URL and, if
so, retrieving the data file from a remote server using the file
identifier and,** if not, retrieving the data file from the data storage
medium using the file identifier; **and**
**manipulating the data file before the data file has been
completely retrieved from the remote server,**

DOCKET NO.: MSFT-0234/155631.1
Application No.: 09/767,768
Advisory Action Dated: September 22, 2004

PATENT
REPLY FILED UNDER EXPEDITED
PROCEDURE PURSUANT TO
37 CFR § 1.116

wherein said file identifier identifies executable code and
retrieving the data file from a remote server comprises transmitting the file
identifier and at least one parameter for executing the executable code.

In order for a reference or set of references to render these claims obvious, the reference must disclose each of the claimed elements, including those emphasized. More particularly, the references must teach the claimed combination including **retrieving a data file from a remote server using the file identifier <u>and manipulating the data file before the data file has been completely retrieved from the remote server.</u>** Applicant's undersigned representative respectfully submits that none of the references even teach the emphasized limitations, and cannot possibly suggest their combination with the other claimed elements.

Morton discloses a general overview of the CGI protocol (Morton, page 1). Morton includes a description of common methods for transmitting data to a CGI script, including the GET and POST methods, as well as using the query string and path info (Id.). In addition, Morton describes how a web browser packages CGI data using url-encoding (Id.).

Breslau et al. purport to disclose a method and apparatus for operating a compiler to process include statements resident at non-connected network locations. (Abstract). When an include option in a source code file specifies an include file resident at a non-connected network, the compiler establishes a connection with the network and downloads the file. (Abstract). The downloaded file is then used to compile the source code. (Abstract). Thus, Breslau et al. teach a system wherein an include file is identified and downloaded from a remote source.

However, neither Morton nor Breslau et al., alone or in combination, teach **retrieving a data file from a remote server using the file identifier <u>and manipulating the data file before the data file has been completely retrieved from the remote server.</u>** Breslau et al. states that "when a remote include file 1301 has been retrieved, it is added to the include file temporary source code file." (Breslau et al., col. 8, ll. 22-24). Breslau et al. further states that "[t]he include file list 1200 of path specifications is processed until all entries 1201 have been processed (step 439), at which point the compilation of the source 104 together with the include file temporary source code file is initiated using well known source code compilation techniques. (Breslau et al., col. 8, ll. 25-30). Thus, according to Breslau et al. processing of

DOCKET NO.: MSFT-0234/155631.1
Application No.: 09/767,768
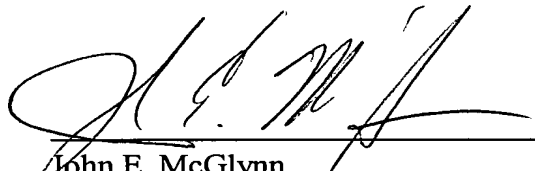Advisory Action Dated: September 22, 2004

PATENT
REPLY FILED UNDER EXPEDITED
PROCEDURE PURSUANT TO
37 CFR § 1.116

the downloaded data does not occur until after all of the data has been downloaded. Conversely, the claims require **manipulating the data file before the data file has been completely retrieved from the remote server**. By teaching that all remote include files be downloaded before starting the compilation process, Breslau et al. actually teaches away from the claimed combination. Accordingly, the combination of Morton, and Breslau et al. cannot possibly render the claims invalid.

Conclusion

Applicant's undersigned representative respectfully requests reconsideration of the claims and early issuance of a Notice of Allowance. If the Examiner does not find the application is in condition for allowance, Applicant's undersigned representative respectfully requests that the Examiner contact him at the telephone number listed below before taking further action.

Date: January 11, 2005

John E. McGlynn
Registration No. 42,863

Woodcock Washburn LLP
One Liberty Place - 46th Floor
Philadelphia PA 19103
Telephone: (215) 568-3100
Facsimile: (215) 568-3439